CLAIMS

1. A method for controlling access to a dynamic random access memory (DRAM),
characterized in that said method comprises the step of performing, for each DRAM access, a sequence of a predetermined number of DRAM control operations in response to a corresponding sequence of control instructions included in microcode instructions of a processor (10).

2. The method according to claim 1,
characterized in that each microcode instruction includes a control instruction, formed by at least one control bit, controlling which one of a plurality of predefined DRAM control operations (R, W, H, E) to perform.

3. The method according to claim 2,
characterized in that said predefined DRAM control operations (R, W, H, E) are arrangeable to form said sequence of DRAM control operations such that a read access, a write access, a page mode read access, a page mode write access, a page mode read write access or a page mode write read access to said DRAM (60) is enabled.

4. The method according to any of the preceding claims,
characterized in that at least one control instruction in said sequence of control instructions temporarily puts the memory cycle of said DRAM (60) on hold.

5. The method according to any of the preceding claims,
characterized in that said method further comprises the step of selecting the cycle time of each microcode instruction from a number of different cycle times such that the cycle time of each microcode instruction matches the duration of the corresponding DRAM control operation.

6. The method according to claim 5,
characterized in that each microcode instruction includes a cycle time control bit determining the cycle time of the microcode instruction, a first logical state of the cycle time control bit indicating a first cycle time and a second logical state of the cycle time control bit indicating a second extended cycle time.

7. The method according to claim 2,
characterized in that a first one, referred to as an R-operation, of said predefined DRAM control operations includes the steps of:

selectively, if inactive, activating a row address strobe (RAS) signal to said DRAM;

selectively enabling a valid row address to be forwarded to said DRAM (60), and, a first predetermined period of time later, enabling a valid column address to be forwarded to said DRAM (60);

selectively, if active, deactivating a write enable (WE) signal to said DRAM; and

selectively, if active, deactivating a column address strobe (CAS) signal to said DRAM, and, a second predetermined period of time later, in the next microinstruction cycle, activating said CAS signal.

8. The method according to claim 2,
characterized in that a second one, referred to as a W-operation, of said predefined DRAM control operations includes the steps of:

selectively, if inactive, activating a row address strobe (RAS) signal to said DRAM;

enabling a valid row address to be forwarded to said DRAM (60), and, a first predetermined period of time later, enabling a valid column address to be forwarded to said DRAM;

selectively, if inactive, activating a write enable (WE) signal to said DRAM; and

selectively, if active, deactivating a column address strobe (CAS) signal to said DRAM, and, a second predetermined period of time later, in the next microinstruction cycle, activating said CAS signal.

9. The method according to claim 2,
characterized in that a third one, referred to as a H-operation, of said predefined DRAM control operations includes the steps of:

deactivating a column address strobe (CAS) signal to said DRAM (60); and

deactivating a write enable signal to said DRAM.

10. The method according to claim 2,
characterized in that a fourth one, referred to as an E-operation, of said predefined DRAM control operations includes the steps of:

deactivating a row address strobe (RAS) signal to said DRAM (60);

selectively, if active, deactivating a column address strobe (CAS) signal to said DRAM;

selectively, if active, deactivating a write enable (WE) signal to said DRAM; and

enabling a valid row address to be forwarded to said DRAM.

11. The method according to claim 7, 9 and 10,
characterized in that, for a read access to said DRAM (60), said sequence of DRAM control operations includes an R-operation, an H-operation and an E-operation, in that order.

12. The method according to claim 8, 9 and 10,
characterized in that, for a write access to said DRAM (60), said sequence of DRAM control operations includes a W-operation, an H-operation and an E-operation, in that order.

13. The method according to claim 7, 9 and 10,
characterized in that, for a page mode read access to said DRAM (60), said sequence of DRAM control operations includes a predetermined number of R-operations followed by an H-operation and an E-operation.

14. The method according to claim 8, 9 and 10,
characterized in that, for a page mode write access to said DRAM (60), said sequence of DRAM control operations includes a predetermined number of W-operations followed by an H-operation and an E-operation.

15. A controller for a dynamic random access memory (DRAM),
characterized in that said DRAM controller (50) is responsive to a sequence of control instructions for controlling access to said DRAM (60), each control instruction being formed by a predetermined part of a microcode instruction of a processor (10).

16. The DRAM controller according to claim 15,
characterized in that said DRAM controller (50) controls access to said DRAM by performing a sequence of a predetermined number of DRAM control operations in response to said sequence of control instructions.

17. The DRAM controller according to claim 15,
characterized in that each control instruction, formed by at least one control bit, controls which one of a plurality of predefined DRAM control operations (R, W, H, E) to perform.

18. The DRAM controller according to claim 16 or 17,
characterized in that the cycle time of each microcode instruction is extendable such that the cycle time of each microcode instruction matches the duration of the corresponding DRAM control operation.

19. The DRAM controller according to claim 18,
characterized in that the cycle time of each microcode instruction is extendable by means of a cycle time control instruction included within the microcode instruction itself.

20. The DRAM controller according to claim 15,
characterized in that at least one control instruction in said sequence of control instructions temporarily puts the memory cycle of said DRAM (60) on hold.

21. The DRAM controller according to claim 15,
characterized in that the microcode instructions of said processor (10) are stored in a program memory (22) separated from said DRAM (60).

22. The DRAM controller according to claim 15,
characterized in that said DRAM controller (50) is responsive to address information, determined by a number of microcode instructions of said processor (10), for addressing said DRAM (60).

23. The DRAM controller according to claim 15,
characterized in that the microcode instructions of said processor (10) are the instructions of a reduced instruction set computing (RISC) processor.

24. A computer system having a processor (10), a primary memory (60) cooperating with said processor, and a memory controller (50) for said primary memory,
characterized in that said memory controller (50) is responsive to a sequence of control instructions from said processor (10) for controlling access to said primary memory (60), each control instruction being formed by a predetermined part of a microcode instruction of said processor (10).

25. The computer system according to claim 24,
characterized in that said primary memory is a DRAM, and said memory controller (50) controls access to said DRAM (60) by performing a sequence of DRAM control operations in response to said sequence of control instructions.

26. The computer system according to claim 25,
characterized in that said processor (10) and said DRAM (60) are provided on the same circuit board.

27. The computer system according to claim 24,
characterized in that said processor (10) is a complex instruction set computing (CISC) processor, and complex instructions are stored in said primary memory (60) and executed by microcode instructions stored in a program memory (22) in said processor (10).

28. The computer system according to claim 25,
characterized in that the cycle time of each microcode instruction is extendable such that the cycle time of each microcode instruction matches the duration of the corresponding DRAM control operation.

29. A method for performing a virtual direct memory access (DMA) to a primary memory (60) in a computer system,
characterized in that said method comprises the steps of:

storing data from/to an input/output device (80) in a buffer (75);

transferring said data between said buffer (75) and said primary memory (60) via internal data paths of a processor (10) of the computer system, said data transfer being controlled by a microcode instruction program (22) of the processor.

30. The method for performing a virtual DMA access to a primary memory according to claim 29,
characterized in that said step of transferring data between said buffer (75) and said primary memory (60) includes the steps of:

transferring data between said buffer (75) and an internal register (55) of said processor (10) in response to control signals generated by said microcode instruction program (22); and

transferring data between said internal register (55) and said primary memory (60) in response to a sequence of control instructions included in microcode instructions of said microcode instruction program (22).

31. The method for performing a virtual DMA access to a primary memory according to claim 30,
characterized in that said primary memory is a dynamic random access memory (DRAM), and said step of transferring data between said internal register (55) and said DRAM includes performing a sequence of DRAM control operations in response to said sequence of control instructions.

32. The method for performing a virtual DMA access to a primary memory according to claim 29,
characterized in that said method further comprises the step of regularly investigating whether a predetermined amount of data is present in said buffer (75) for inputs to the primary memory (60), and whether there is a predetermined amount of free space available in said buffer (75) for outputs from the primary memory (60), said transfer between said buffer and said primary memory being initiated in dependence upon the outcome of said investigation.

33. The method for performing a virtual DMA access to a primary memory according to claim 32,
characterized in that said investigating step is performed by at least one microcode instruction that is activated at a predetermined frequency.

34. The method for performing a virtual DMA access to a primary memory according to claim 29,
characterized in that said method further comprises at least one of processing and monitoring, in said processor (10), of data transferred between said buffer and said primary memory (60) via said internal data paths of said processor.

35. The method for performing a virtual DMA access to a primary memory according to claim 34,
characterized in that said processing comprises at least one of the following: data conversion, data encoding, data decoding, image data compression, image data decompression, scaling, pattern matching and checksum calculation.

36. A computer system having a processor (10) and a primary memory (60) coupled to said processor,
characterized in that said computer system further comprises:
a buffer (75) for storing data from/to an input/output device (80); and
means for transferring said data between said buffer (75) and said primary memory (60) via internal data paths of the processor (10) under the control of a microcode instruction program (22) in the processor.

37. The computer system according to claim 36,
characterized in that said means for transferring data between said buffer (75) and said primary memory (60) includes:
means for transferring data between said buffer (75) and an internal register (55) of said processor (10) in response to control signals generated by said microcode instruction program (22); and
means for transferring data between said internal register (55) and said primary memory (60) in response to a sequence of control instructions included in microcode instructions of said microcode instruction program (22).

38. The computer system according to claim 37, characterized in that said means for transferring data between said buffer (75) and said internal register (55) includes a DMA controller (70), which also controls transfer of data between said input/output device (80) and said buffer (75).

39. The computer system according to claim 37, characterized in that said primary memory (60) is a dynamic random access memory (DRAM), and said means for transferring data between said internal register (55) and said DRAM includes a DRAM controller (50) for performing a sequence of DRAM control operations in response to said sequence of control instructions.

40. The computer system according to claim 36, characterized in that said microcode instruction program (22) of said processor (10) is configured for performing at least one of processing and monitoring of data transferred between said buffer (75) and said primary memory (60) via the internal data paths of said processor.

41. The computer system according to claim 40, characterized in that said processing comprises at least one of the following: data conversion, data encoding, data decoding, image data compression, image data decompression, scaling, pattern matching and checksum calculation.